SUMO

Session - 1

Basic About **SUMO**

What is SUMO?

Simulation of **U**rban **MO**bility

Traffic simulator

Open source

Microscopic

Continuous, multi-modal traffic simulation

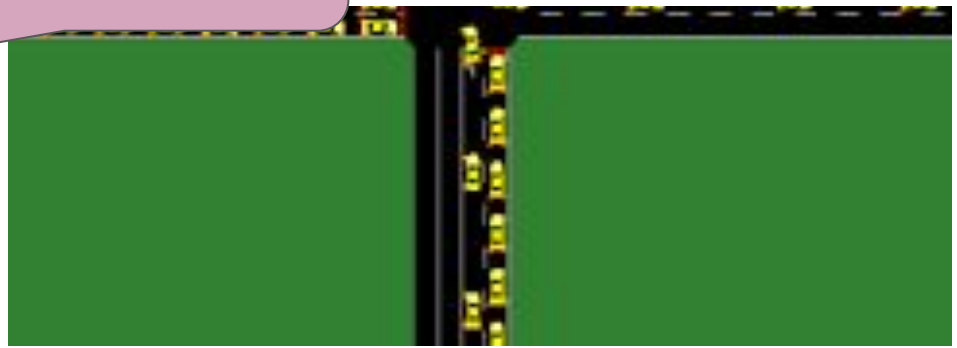
Simulate the movement of individual vehicles based on car-following and lane-changing theories

Two or more different modes of transport

main · 11 branches · 50 tags · Go to file · Code

namdre #2 doc #12043 · ed56559 11 hours ago · 29,018 commits

.github	adapting action fix #11953	12 days ago
bin	delete invalid net in bin folder	2 years ago
build	updating fedora docker #3	2 days ago
data	NEMA defaults to rightOfWay=allwayStop (when switched off). refs #12…	11 hours ago
docs	#2 doc #12043	11 hours ago
src	NEMA defaults to rightOfWay=allwayStop (when switched off). refs #12…	11 hours ago
tests	updated network with new off-behavior (not expecting change in sim re…	11 hours ago
tools	more helpful error background images message (#12044)	20 hours ago
unittest	removing unused function with insecure buffer operations #12	2 months ago
.appveyor.yml	more appveyor speedup #10689	6 months ago
.gitattributes	changing to UTF8, fixing rename on windows #11237	last month
	preparing gettext infra #11237	last month
	removing sumolib4matlab submodule due to unclear license	3 years ago
	updating authors #2	3 months ago
cff	version bump 1.15.0 #563	9 days ago



Who developed SUMO?

Developed by employees of the **Institute of Transportation Systems** at the **German Aerospace Center - (founded in 1969)**

- Deutsches Zentrum für Luft- und Raumfahrt e.V. **(DLR)**

Why SUMO? Why not others

Open source - **EPL 2.0**

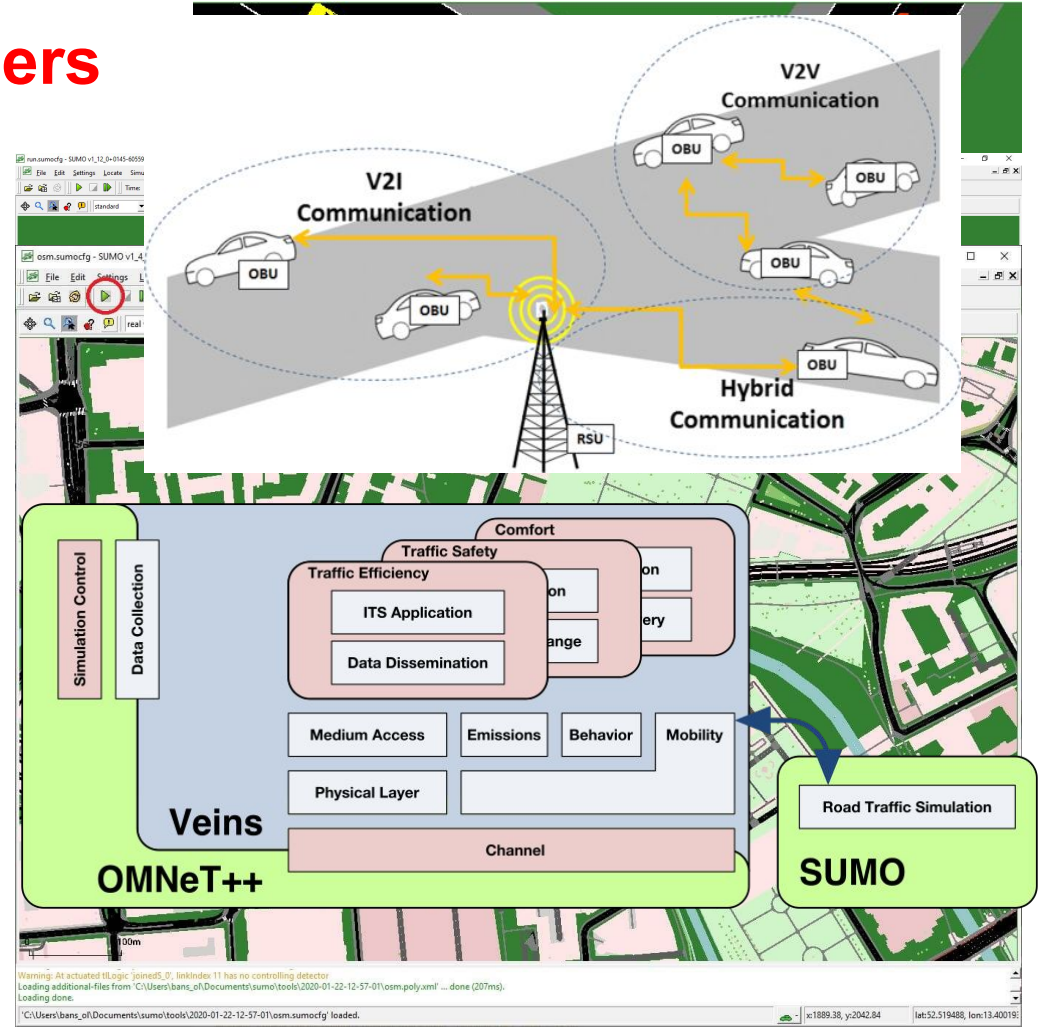
Different vehicle types

Multi-lane streets with lane changing

Imports VISUM, Vissim, Shapefiles, **OSM**,

Simulation of vehicular communications

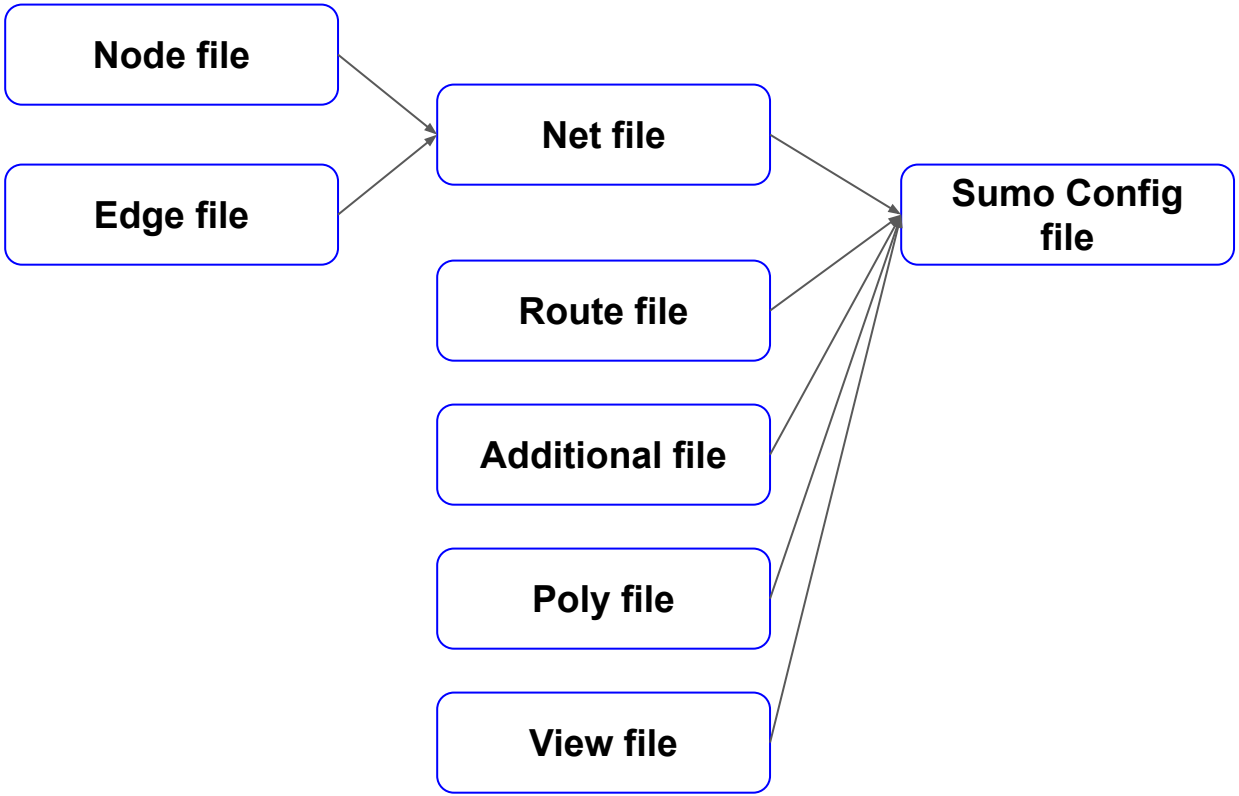
More…



Installation of **SUMO**

https://sumo.dlr.de/docs/Installing/index.html

Basic files SUMO


Node file
Edge file
Net file
Route file
Additional file
Poly file
View file
Sumo Config file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<nodes xmlns:xsi="http://www.w3.org/2001/X
    <node id="1" x="-250.0" y="0.0" />
    <node id="2" x="+250.0" y="0.0" />
    <node id="3" x="+251.0" y="0.0" />
</nodes>
```

Node file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<edges xmlns:xsi="http://www.w3.org/2001
    <edge from="1" id="1to2" to="2" />
    <edge from="2" id="out" to="3" />
</edges>
```

Edge file

```xml
<edge id=":2_0" function="internal">
    <lane id=":2_0_0" index="0" speed="13.90" length="0.10" shape="500.00,-1.65 500.00,-1.65"/>
</edge>

<edge id="1to2" from="1" to="2" priority="-1">
    <lane id="1to2_0" index="0" speed="13.90" length="500.00" shape="0.00,-1.65 500.00,-1.65"/>
</edge>
<edge id="out" from="2" to="3" priority="-1">
    <lane id="out_0" index="0" speed="13.90" length="1.00" shape="500.00,-1.65 501.00,-1.65"/>
</edge>

<junction id="1" type="unregulated" x="0.00" y="0.00" incLanes="" intLanes="" shape="-0.00,-0.05 -0.00,-3.25"/>
<junction id="2" type="priority" x="500.00" y="0.00" incLanes="1to2_0" intLanes=":2_0_0" shape="500.00,-0.05 500.00,-3.25 500.00,-0.05">
    <request index="0" response="0" foes="0" cont="0"/>
</junction>
<junction id="3" type="unregulated" x="501.00" y="0.00" incLanes="out_0" intLanes="" shape="501.00,-3.25 501.00,-0.05"/>

<connection from="1to2" to="out" fromLane="0" toLane="0" via=":2_0_0" dir="s" state="M"/>

<connection from=":2_0" to="out" fromLane="0" toLane="0" dir="s" state="M"/>
```

Net file


```xml
<?xml version="1.0" encoding="UTF-8"?>
<routes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  <vType accel="1.0" decel="5.0" id="Car" length="5.0" minGap=

  <route id="route0" edges="1to2 out" />

  <vehicle depart="1" id="veh0" route="route0" type="Car" />
</routes>
```

Route file

```xml
<input>
    <net-file value="hello.net.xml"/>
    <route-files value="hello.rou.xml"/>
</input>
```

Config file

Run **SUMO**

----- How to create simple network and make one car to route on road ----------

1: Please go to this website: https://github.com/eclipse/sumo/tree/main/tests/complex/tutorial/hello/data

2: Copy node file and edge file (to some folder)
simple.edg.xml
simple.nod.xml

3: Open cmd in your windows

4: Go to the folder that you saved all files (simple.edg.xml,simple.nod.xml)

5: Type this command
> netconvert --node-files=simple.nod.xml --edge-files=simple.edg.xml --output-file=simple.net.xml

6: Now you can see in the folder one more file named simple.net.xml

7: From website copy two more files -- hello.rou.file and hello.sumocfg
simple.rou.xml
simple.sumocfg

8: Open the simple.sumocfg file and remove

```
<time>
    <begin value="0"/>
    <end value="10000"/>
</time>
<gui_only>
    <gui-settings-file value="hello.settings.xml"/>
</gui_only>
```

9: Change the contents like this and save it.(simple.sumocfg)

```
<input>
<net-file value="simple.net.xml"/>
<route-files value="simple.rou.xml"/>
</input>
```

10: From the terminal or command

```
> sumo-gui -c simple.sumocfg
```

11: Click the run button once you see the SUMO gui.

If one car will run on the simulation, That's all You have completed the first hello world program in SUMO

If you cannot see the car, then set delay button to 200 seconds.

Using NETEDIT **SUMO**

SUMO

Session - 2

24-11-2022

Outline

Basic Python

SUMO Loop and Area Detectors

Generate Route files from Python

Control traffic Light using Python

Calculate waiting time and other measurements

OSM - **O**pen **S**treet **M**ap and Random Route

Multivesta with SUMO

Outline

Basic Python

SUMO Loop and Area Detectors

Generate Route files from Python

Control traffic Light using Python

Calculate waiting time and other measurements

OSM - **O**pen **S**treet **M**ap and Random Route

Multivesta with SUMO

Basic **Python**

Hello World

```
# first Program

print("Hello World!!!")
```

Add Two Numbers

```
# This program adds two numbers

num1 = 1.5
num2 = 6.3

# Add two numbers
sum = num1 + num2

# Display the sum
print("Sum: ", sum)
#print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))
#print('The sum of %s and %s is %s'%(num1, num2, sum))
```

Add Two Numbers With User Input

```
# This program adds two numbers

# Store input numbers
num1 = int(input('Enter first number: '))   int()
num2 = input('Enter second number: ')

# Add two numbers
sum = num1 + num2

# Display the sum
print("Sum: ", sum)
#print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))
#print('The sum of %s and %s is %s'%(num1, num2, sum))
```

Find the Square Root

```
# Python Program to calculate the square root

# Note: change this value for a different result
num = 8

# To take the input from the user
#num = float(input('Enter a number: '))

num_sqrt = num ** (1/3)
print(“Square root: ”, num_sqrt)
print('The square root of %0.3f is %0.3f'%(num ,num_sqrt))
```

Find the Square Root

```
# Python Program to calculate the square root
import math
# Note: change this value for a different result
num = 8

# To take the input from the user
#num = float(input('Enter a number: '))

num_sqrt = math.sqrt(num)
print(“Square root: ”, num_sqrt)
print('The square root of %0.3f is %0.3f'%(num ,num_sqrt))
print(math.sqrt(num))
```

Swap the two numbers

```
# Python program to swap two variables

x = 5
y = 10

# To take inputs from the user
#x = input('Enter value of x: ')
#y = input('Enter value of y: ')

# create a temporary variable and swap the values
temp = x
x = y
y = temp

print('The value of x after swapping: {}'.format(x))
print('The value of y after swapping: {}'.format(y))
```

Swap the two numbers

```
# Python program to swap two variables

x = 5
y = 10

# To take inputs from the user
#x = input('Enter value of x: ')
#y = input('Enter value of y: ')

# create a temporary variable and swap the values
temp = x
x = y
y = temp

print('The value of x after swapping: {}'.format(x))
print('The value of y after swapping: {}'.format(y))
```

Swap the two numbers

```
# Program to generate a random number between 0 and 9

# importing the random module
import random

print(random.randint(0,9))


print(“x = ” , x)
print(“y = ”, y)
```

Python Looping Techniques

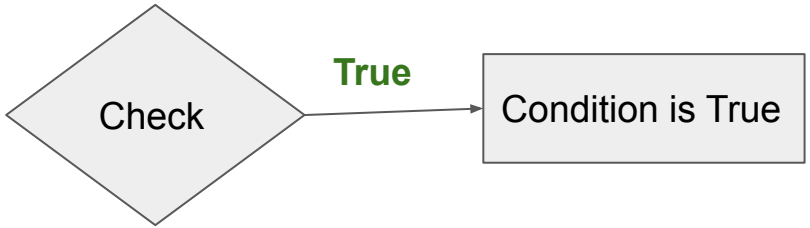
If

For

While

Python Looping Techniques

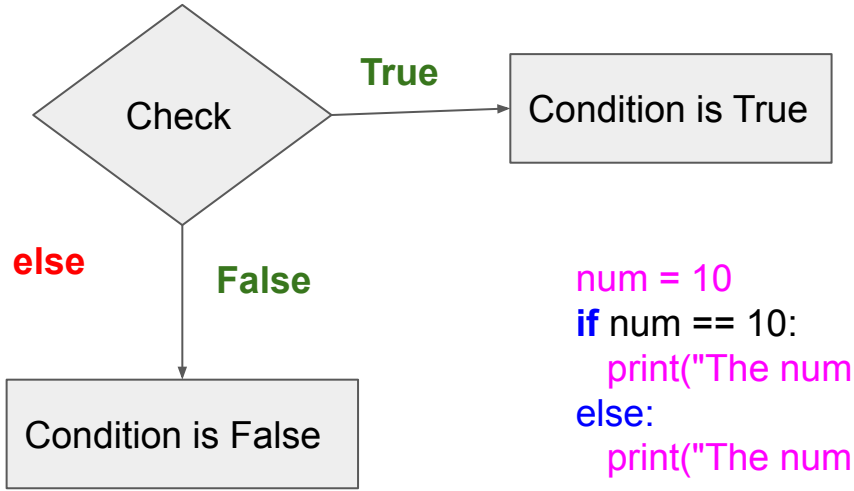
If




```
num = 10
if num == 10:
   print("The number is 10")
```

Python Looping Techniques

If - else




```
num = 10
if num == 10:
   print("The number is 10")
else:
   print("The number is not 10")
```

Python Looping Techniques

For

```
for i in range(10):
    print(i)
```

```
for i in range(0,10,2):
    print(i)
```

```
for i in range(10,0,-1):
    print(i)
```

```
num = [1,4,7,8,13]
for i in range(len(num)):
    print(num[i])
```

Python Looping Techniques

While

```
while True:
    num = int(input("Enter an integer: "))
    print("The double of",num,"is",2 * num)

i = 0
while i < 10:
    print( i )
    i = i +1

i = 0
while i < 10:
    print( i )
    i += 1
```

```
# import random module
import random

while True:
    input("Press enter to roll the dice")

    # get a number between 1 to 6
    num = random.randint(1,6)
    print("You got",num)
    option = input("Roll again?(y/n) ")

    # condition
    if option == 'n':
        break
```

Python Looping Techniques

```
 with open("data/sample.txt", "w") as routes:
  for i in range(10):
    print(i,file=routes)
```

```
with open("data/sample.txt", "w") as routes:
 for i in range(10):
   print('    <vehicle id="right_%i" type="typeWE" route="right" depart="%i" />' % (vehNr, i), file=routes)
   vehNr += 1
```

Outline

Basic Python

SUMO Loop and Area Detectors

Generate Route files from Python

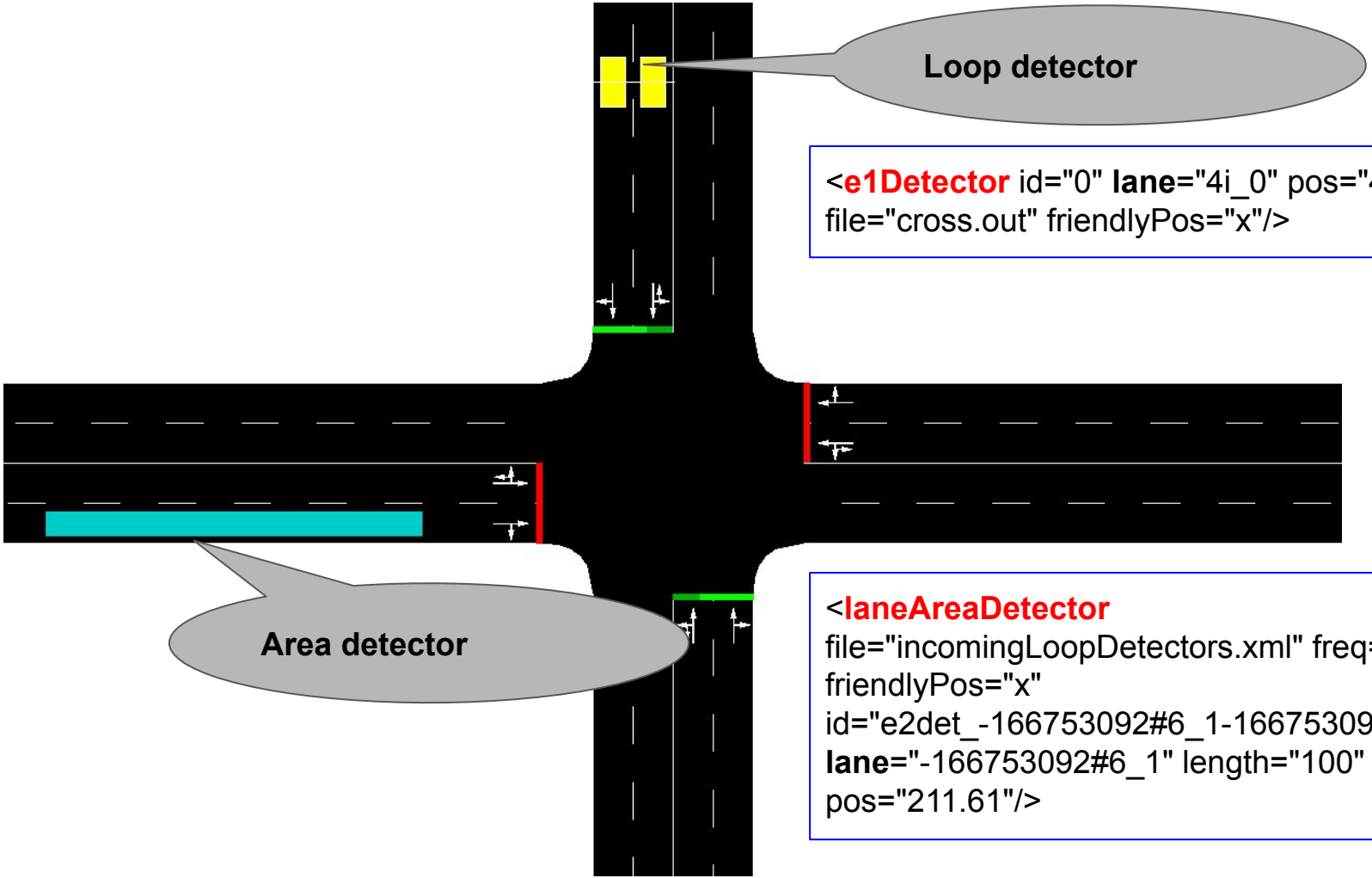
Control traffic Light using Python

Calculate waiting time and other measurements

OSM - **O**pen **S**treet **M**ap and Random Route

Multivesta with SUMO

SUMO Loop and Area Detectors




```
<e1Detector id="0" lane="4i_0" pos="450" freq="30"
file="cross.out" friendlyPos="x"/>
```

```
<laneAreaDetector
file="incomingLoopDetectors.xml" freq="60"
friendlyPos="x"
id="e2det_-166753092#6_1-166753092#6_1"
lane="-166753092#6_1" length="100"
pos="211.61"/>
```

1: https://github.com/eclipse/sumo/tree/main/tests/complex/tutorial/traci_tls

2: Go to data folder, copy the cross.det.xml

SUMO

Session - 2

Outline

Basic Python

SUMO Loop and Area Detectors

Generate Route files from Python

Control traffic Light using Python

Calculate waiting time and other measurements

OSM - **O**pen **S**treet **M**ap and Random Route

Multivesta with SUMO

Outline

Basic Python

SUMO Loop and Area Detectors

Generate Route files from Python

Control traffic Light using Python

Calculate waiting time and other measurements

OSM - Open **S**treet **M**ap and Random Route

Multivesta with SUMO

Multivesta with SUMO

1: Check if SUMO is running using python

https://github.com/eclipse/sumo/tree/main/tests/complex/tutorial/traci_tls

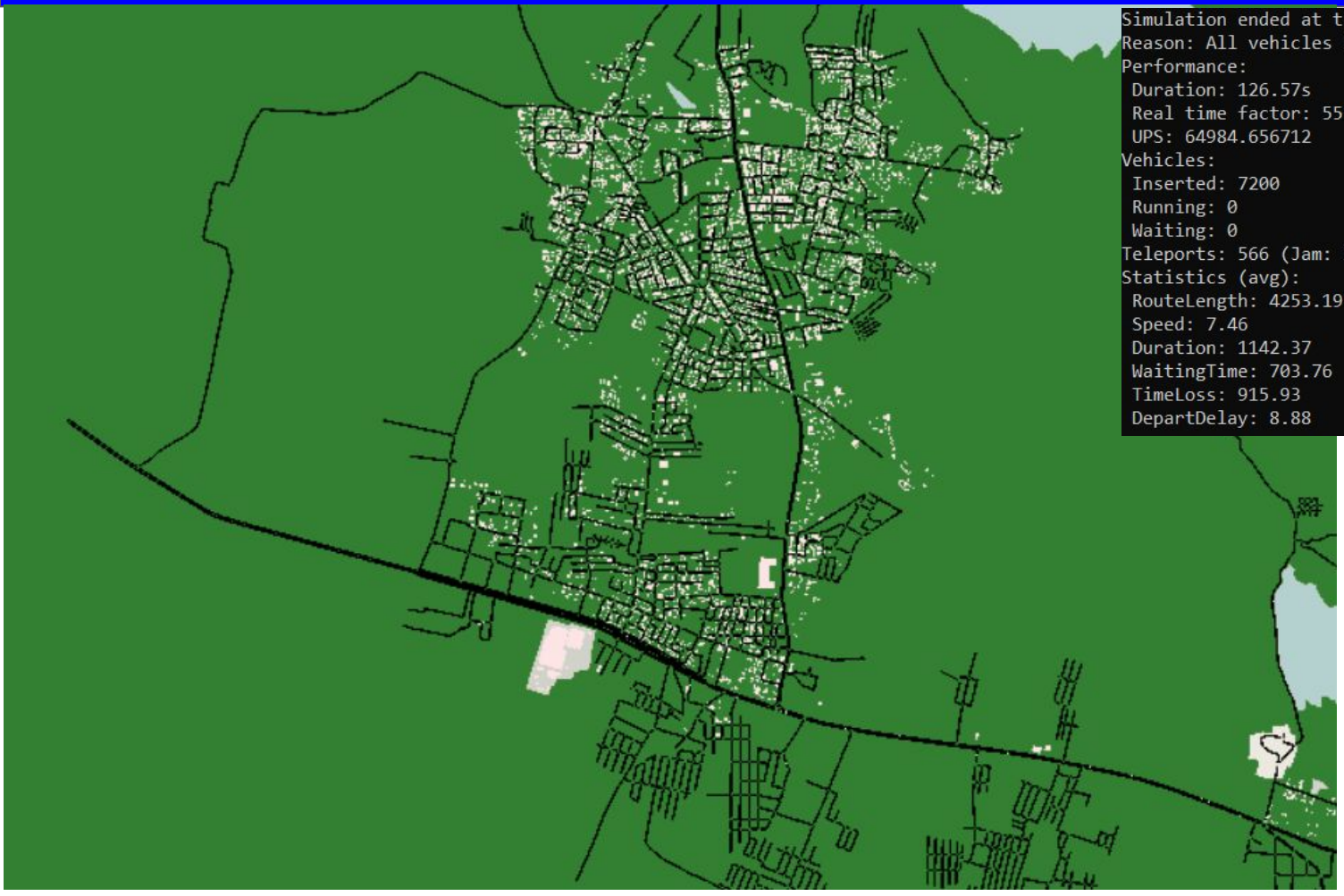
```
# this is the main entry point of this script
if __name__ == "__main__":
    options = get_options()

    # this script has been called from the command line. It will start sumo as a
    # server, then connect and run
    if options.nogui:
        sumoBinary = checkBinary('sumo')
    else:
        sumoBinary = checkBinary('sumo-gui')

    # first, generate the route file for this simulation
    generate_routefile()

    # this is the normal way of using traci. sumo is started as a
    # subprocess and then the python script connects and runs
    traci.start([sumoBinary, "-c", "data/cross.sumocfg",
                             "--tripinfo-output", "tripinfo.xml"])
    run()
```

Motivation

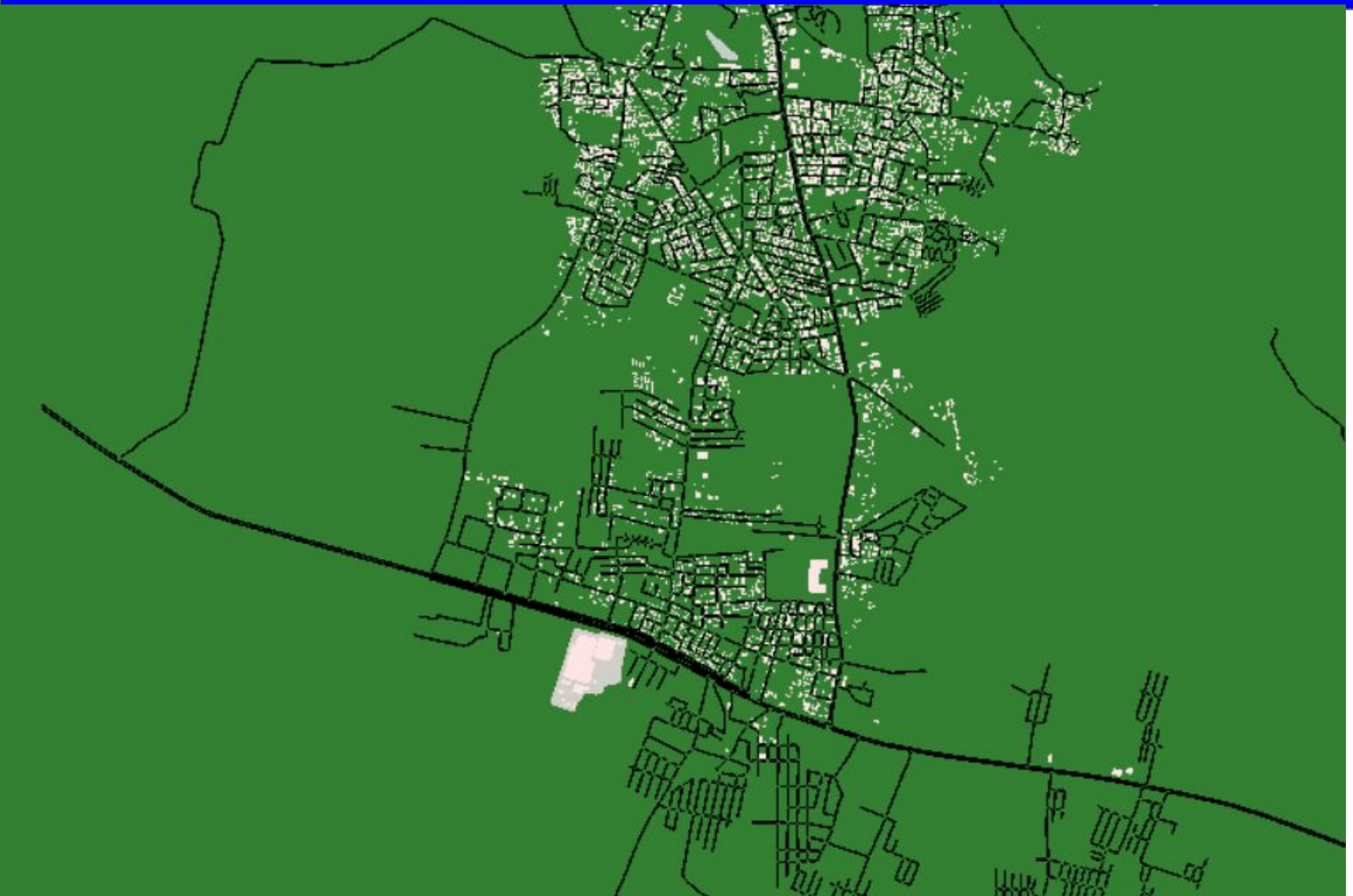


```
Simulation ended at time: 7064.00
Reason: All vehicles have left the simulation.
Performance:
 Duration: 126.57s
 Real time factor: 55.811
 UPS: 64984.656712
Vehicles:
 Inserted: 7200
 Running: 0
 Waiting: 0
Teleports: 566 (Jam: 163, Yield: 226, Wrong Lane: 177)
Statistics (avg):
 RouteLength: 4253.19
 Speed: 7.46
 Duration: 1142.37
 WaitingTime: 703.76
 TimeLoss: 915.93
 DepartDelay: 8.88
```

Simple query

What is the average waiting time or delay?

Motivation



Complex query

What is the probability that emergency vehicle reached near hospital within 10 min under given traffic condition?

What is the traffic volume at Intersection-1 which causes the deadlock at Intersection-2?

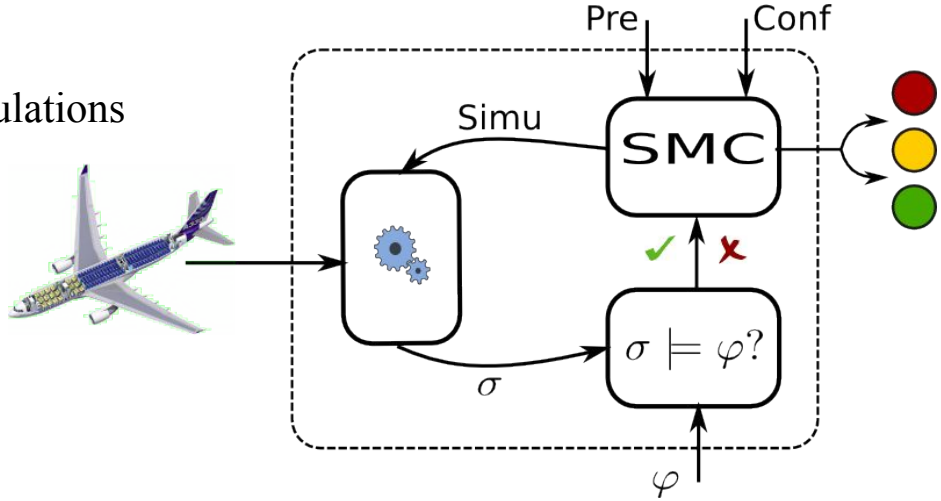
(Statistical) Model Checking [1]

combines simulation and statistical methods for the analysis of stochastic systems

Model + (temporal) logic (e.g PCTL)

Monte-carlo sampling of the runs of the system

Chernoff hoeffding for confidence interval calculations



1 - Legay A., Delahaye B., Bensalem S. (2010) Statistical Model Checking: An Overview. In: Barringer H. et al. (eds) Runtime Verification. RV 2010. Lecture Notes in Computer Science, vol 6418. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-16612-9_11

Car-following models

A method used to determine how vehicles follow one another on a roadway

- higher the speed of the vehicle, higher will be the spacing between the vehicles
- safe distance

- Krauss [1]
- Wiedemann [2]
- Intelligent Driver Model [3]

Lane-changing models

1: Krauß, S., Wagner, P., Gawron, C.: Metastable states in a microscopic model of traffic flow. Physical Review E 55(5), 5597 (1997)

2: Wiedemann, R.: Simulation des strassenverkehrsflusses. Institut fur Verkehrswesen der Universitat Karlsruhe (1994)

3: Treiber, M., Hennecke, A., Helbing, D.: Congested traffic states in empirical observations and microscopic simulations. Physical review E 62(2), 1805 (2000)

Lane-changing models

The subject vehicle in the current lane tries to change direction either to its left or to its right

- If the gap in the selected lane is acceptable the lane change occurs or else it will remain in the current lane

- LC2013 [1]
- SL2015 [2]

1: Mintsis, E., Koutras, D., Porfyri, K., Mitsakis, E., L¨ucken, L., Erdmann, J., Fl¨otter¨od, Y.P., Alms, R., Rondinone, M., Maerivoet, S., Carlier, K., Zhang, X., Blokpoel, R., Harmenzon, M., Boerma, S.: Transaid deliverable 3.1 – modelling, simulation and assessment of vehicle automations and automated vehicles' driver behaviour in mixed traffic (09 2019)

2: Erdmann, J.: Sumo's lane-changing model. In: Modeling Mobility with Open Data, pp. 105–123. Springer (2015)

Tools

MultiVeStA [1]

Statistical Model Checking tool - from VeStA [3] family

Support direct integration with discrete time simulator

QUAntitative Temporal EXpressions language (QuaTEx) - **express systems properties ,supports PCTL,CSL.**

SUMO - *Simulation of Urban Mobility [2]*

Microscopic traffic simulator

Support online interaction through **Traci**

Support several car-following and lane-changing models

1 - Sebastio, Stefano, and Andrea Vandin. "MultiVeStA: Statistical model checking for discrete event simulators." (2013): 1-10.

2 - Krajzewicz, Daniel, Jakob Erdmann, Michael Behrisch, and Laura Bieker. "Recent development and applications of SUMO-Simulation of Urban MObility." International journal on advances in systems and measurements 5, no. 3&4 (2012).

3: Sen, Koushik & Viswanathan, Mahesh & Agha, Gul. (2005). VESTA: A statistical model-checker and analyzer for probabilistic systems. QEST 2005 - Proceedings Second International Conference on the Quantitative Evaluation of SysTems. 2005. 251 - 252. 10.1109/QEST.2005.42.

Multivesta with SUMO

MultiVeStA is an efficient ***statistical analysis tool*** which

Can be easily integrated with existing discrete-event simulators and agent-based models

Enriching them with automated statistical analysis techniques from the family of Statistical Model Checking

Allows to distribute simulations in the cores of a machine or in a network *for free*

Multivesta with SUMO

Transient analysis: what is the expected value of a model's property at a given point in time/as time progresses?

Counterfactual analysis: do the model dynamics change significantly across different parameterizations?

Steady-state analysis: what is the expected value of a model's property on the long run / after it stabilizes?

Ergodicity diagnosis: does the model actually has a steady-state? / does it make sense at all to perform a steady-state analysis?

Integration of Tools

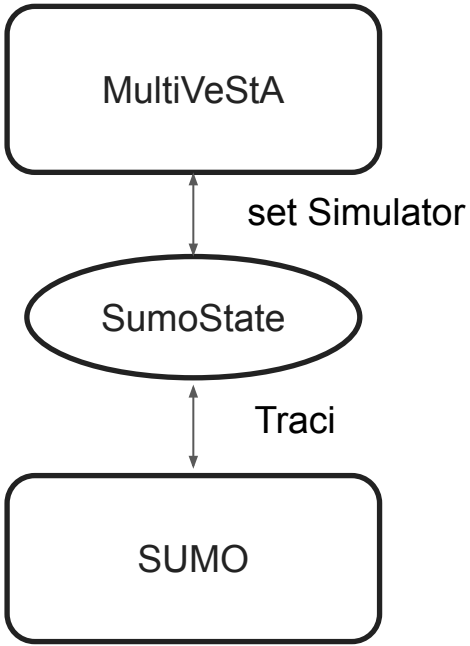
MultiVeStA + SUMO

setSimulatorForNewSimulation(randomSeed)

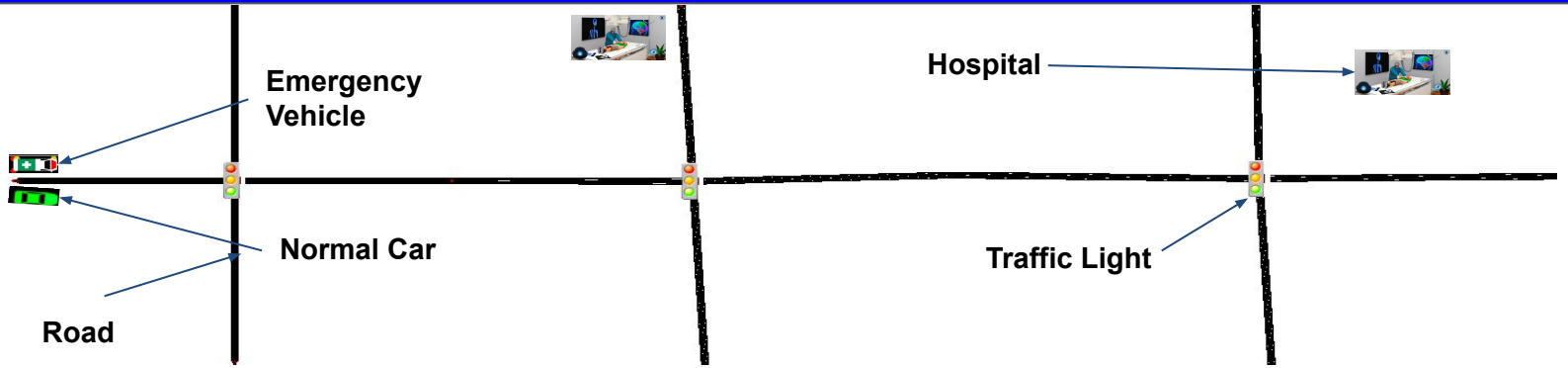
performOneStepOfSimulation()

rval(int)

- rval(0) - the current time
- rval(3) - the number of cars waiting
- rval(4) - the time loss of vehicle
- rval(6) - number of vehicles that reach their destination
- rval(7) - the CO2 emission



Results



Road network with hospital and emergency vehicle

```
-m data / cross . sumocfg
-l serversLists / oneLocalServer
-f quatex / exper1 . quatex
- bs 30 -a 0.1 - d1 x
// x = 2 for queries for non probabilistic operator
// x = 0.1 for these queries probabilistic operator
- vp TRUE
- osws ONESTEP - sots 0 - sd sumoState
```

Parameters of MultiVeStA Client

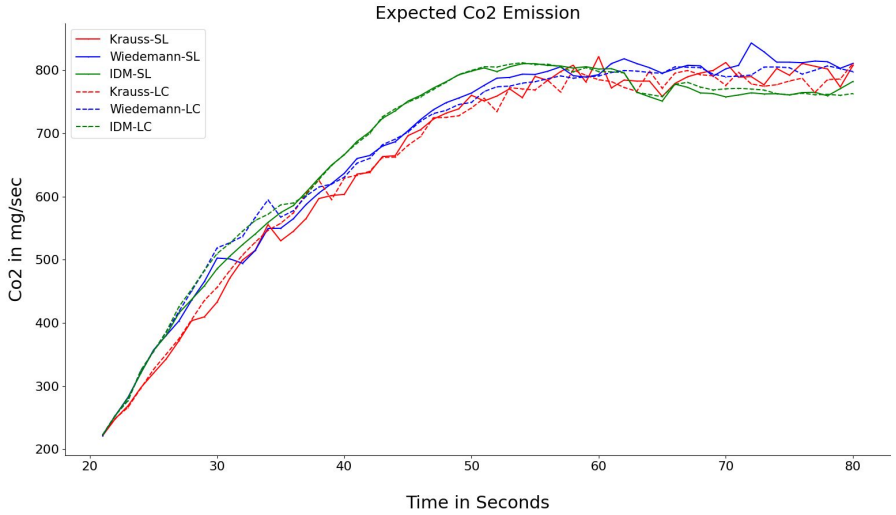
Results

Simple Query:

Estimate expected CO2 emissions within simulation time.

expCo2Emission(x) = if (s. rval (0) >= x) then (s.rval(7))
else # **expCo2Emission** ((x)) fi ;

eval parametric(E[**expCo2Emission** ((k))], k,1.0,1.0 ,100.0) ;



Expected CO_2 for various models

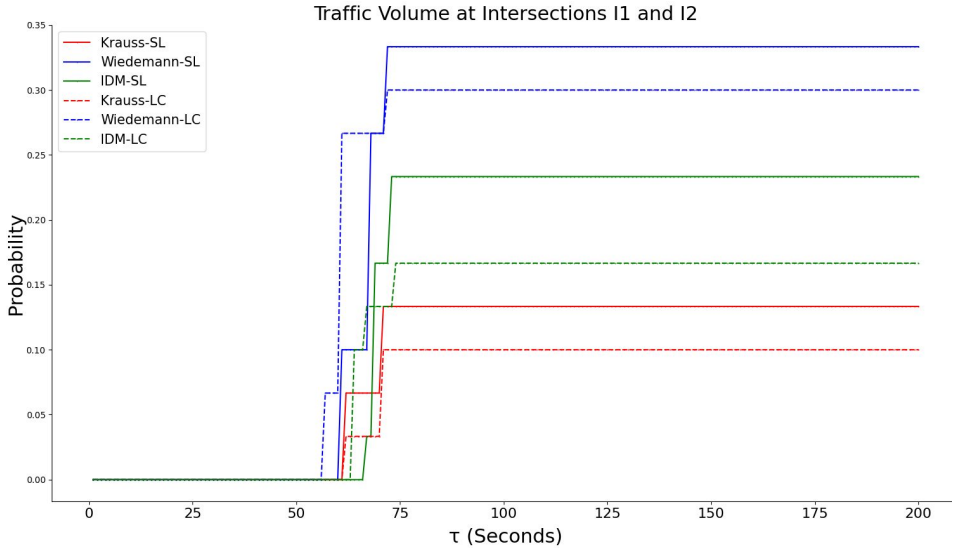
- Krauss
- Wiedemann
- Intelligent Driver Model
- LC2013
- SL2015

Results

Complex Query:

The traffic volume at intersection I1 is less **until** the point the traffic volume is high at the intersection I2

```
t1Ut2(k,x,y) = if( s.rval (0) <= k)
                    then if ( s.rval(11) > x )
                            then (1)
                         else if ( s.rval (10) <= y )
                            then #t1Ut2((k),(x),(y))
                         else (0) fi fi
                 else (0) fi ;
eval parametric(E[ t1Ut2((k) ,(20),(15)) ],
k, 1.0, 1.0, 200.0);
```



Probability that the "traffic volume" at I1 is less than 15 Until the traffic volume at I2 is greater than 20.

Thamilselvam, B., Kalyanasundaram, S., Parmar, S., Panduranga Rao, M.V. (2021). **Statistical Model Checking for Traffic Models.** In: Campos, S., Minea, M. (eds) Formal Methods: Foundations and Applications. SBMF 2021. Lecture Notes in Computer Science(), vol 13130. Springer, Cham. https://doi.org/10.1007/978-3-030-92137-8_2

Multivesta with SUMO

2: Go to this page and download the “MultivestaSumoPythonVersion”

https://github.com/ThamilselvamB/Multivesta-With-SUMO

Slides
https://docs.google.com/presentation/d/1KzHpDUooLlhAgtZ8J5TAsjReNs8KaCecEHy3-hz-9_c/edit?usp=sharing

Github link: www.github.com/thamilselvamB

Youtube Channel
https://www.youtube.com/@vnyinstitute7076